

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

UNITED STATES PATENT APPLICATION

of

DOUGLAS DILLON

for

APPARATUS AND METHOD FOR HYBRID NETWORK ACCESS

LAW OFFICES

NEGAN, HENDERSON
ARABOW, GARRETT
& DUNNER

1300 I STREET, N. W.
WASHINGTON, DC 20005

1-202-408-4000

Background of the Invention

This application relates to a computer network and, more specifically, to a method and apparatus for allowing both high-speed and regular-speed access to a computer network.

10 The Internet is an example of a TCP/IP network. The Internet has over 10 million users. Conventionally, access to the Internet is achieved using a slow, inexpensive method, such as a terrestrial dial-up modem using a protocol such as SLIP (Serial Line IP), PPP, or by using a fast, more expensive method, such as
15 a switched 56 Kbps, frame relay, ISDN (Integrated Services Digital Network), or T1.

 Users generally want to receive (download) large amounts of data from networks such as the Internet. Thus, it is desirable to have a one-way link that is used only for downloading information
20 from the network. A typical user will receive much more data from the network than he sends. Thus, it is desirable that the one-way link be able to carry large amounts of data very quickly. What is needed is a high bandwidth one-way link that is used only for downloading information, while using a slower one-way link is used
25 to send data into the network.

 Currently, not all users have access to high speed links to networks. Because it will take a long time to connect all users to networks such as the Internet via physical high-speed lines, such as fiber optics lines, it is desirable to implement some type of high-speed line that uses the existing infrastructure.

5 Certain types of fast network links have long propagation delays. For example, a link may be transmitting information at 10 Mbps, but it may take hundreds of milliseconds for a given piece of information to travel between a source and a destination on the network. In addition, for even fast low-density links, a slow speed return-link may increase the round trip propagation time, and thus limit throughput. The TCP/IP protocol, as commonly implemented, is not designed to operate over fast links with long propagation delays. Thus, it is desirable to take the propagation delay into account when sending information over such a link.

15 Summary of the Invention

The present invention overcomes the problems and disadvantages of the prior art by allowing a user to download data using a fast one-way satellite link, while using a conventional low-speed Internet connection for data being sent into the network. The invention uses a "spoofing" technique to solve the problem of the long propagation delays inherent in satellite communication.

20 In accordance with the purpose of the invention, as embodied and broadly described herein, the invention is a network system that forms a part of a network, comprising: a source computer, having a link to the network; a destination computer, having a link to the network; a satellite interface between the source computer and the destination computer, wherein information passes from the source computer to the destination computer; means in the destination computer for requesting information from the source computer over the network; means for receiving an information packet sent from the source computer in response to the request

5 and for sending the information packet to the destination computer over the satellite interface; and means for sending an ACK message to the source computer in response to receipt of the information packet, wherein the ACK message appears to the source computer to have come from the destination computer.

10 In further accordance with the purpose of the invention, as embodied and broadly described herein, the invention is a gateway in a network system that forms a part of a TCP/IP network, wherein the network includes a source computer having a link to the TCP/IP network and a link to a high speed satellite interface, and a
15 destination computer having a link to the TCP/IP network and a link to the high speed satellite interface, the gateway comprising: means for receiving an information packet sent from the source computer and for sending the information packet to the destination computer over the satellite interface; and means for
20 sending an ACK message to the source computer in response to receipt of the information packet, wherein the ACK message appears to the source computer to have come from the destination computer.

25 Objects and advantages of the invention will be set forth in part in the description which follows and in part will be obvious from the description or may be learned by practice of the invention. The objects and advantages of the invention will be realized and attained by means of the elements and combinations particularly pointed out in the appended claims.

5

Brief Description of the Drawings

The accompanying drawings, which are incorporated in and constitute a part of this specification, illustrate several embodiments of the invention and, together with the description, serve to explain the principles of the invention.

10

Fig. 1 is a hardware block diagram of a preferred embodiment of the invention;

Fig. 2 is a diagram of a portion of a hybrid terminal of Fig. 1;

Fig. 3 is a diagram showing an IP packet format;

15 Fig. 4 is a diagram showing a plurality of packet formats, including an Ethernet packet format;

Fig. 5 is a diagram showing a tunnelling packet format;

Fig. 6 is a diagram of steps performed by the hybrid terminal of Fig. 1;

20 Fig. 7 is a diagram showing an example of partial data in a tunnelling packet;

Fig. 8 is a flowchart of steps performed by the hybrid terminal of Fig. 1;

25 Fig. 9 is a diagram of steps performed by a hybrid gateway of Fig. 1;

Fig. 10 is a diagram showing a format of packets sent to a satellite gateway of Fig. 1;

Fig. 11 is a diagram showing a TCP packet format;

Fig. 12 is a ladder diagram showing packets sent from an application server to the hybrid gateway and from the hybrid gateway to the hybrid terminal over a satellite link; and

5 Figs. 13(a) through 13(e) are flowcharts of steps performed by the hybrid gateway of Fig. 1.

Detailed Description of the Preferred Embodiments

Reference will now be made in detail to the preferred embodiments of the invention, examples of which are illustrated in the accompanying drawings. Wherever possible, the same reference numbers will be used throughout the drawings to refer to the same or like parts.

a. General Overview

10 A preferred embodiment of the present invention uses satellite technology to implement a high-speed one way link between a user's computer and a TCP/IP network, such as the Internet or a private TCP/IP network. This high-speed link is used to download data from the network. The user's computer also has a conventional TCP/IP link for sending data to the network. The invention can use various forms of high-speed, one-way links, such as satellites, and cable television lines. The invention can use various forms of low-speed networks, such as TCP/IP networks, dialup telephones, ISDN D-channel, CPDP, and low-speed satellite paths.

20 The described embodiment of the present invention uses satellites to provide a high-speed one-way link. Satellites can cover large geographical areas and are insensitive to the distance between a transmitter and a receiver. In addition, satellites are very efficient at point-to-point and broadcast applications, and are resilient and resistant to man-made disasters. Two-way satellites are expensive to use, however, because of the costs.

5 involved in purchasing and installing satellite earth station hardware. In the past, these costs have placed satellite communications outside the reach of the consumer.

The present invention allows a personal computer to receive downloaded information from the network via a satellite at a very practical cost. In the present invention, the cost of satellite communications is reduced because a one-way satellite link is used. Receive-only earth station equipment is cheaper to manufacture because it requires less electronics than send/receive antennae.

15 As is well-known in the art, communication over the Internet and similar TCP/IP networks is achieved through a group (suite) of protocols called Transmission Control Protocol/Internet Protocol (TCP/IP). The TCP/IP protocol is described in the book "Internetworking With TCP/IP, Vol I" by Douglas Comer, published by Prentice-Hall, Inc., of Englewood Cliffs, N.J., 1991, which is incorporated by reference.

20 b. Hybrid TCP/IP Access

Fig. 1 is a hardware block diagram of a preferred embodiment of the invention. Fig. 1 includes five subsystems: a hybrid terminal 110, a SLIP provider (Internet connection) 130, an application server 140, a hybrid gateway 150, and a satellite gateway 160. Hybrid terminal 110 is connected to a modem 190, e.g., a 9600 baud modem, which connects to SLIP provider 130 through a telephone line 192. A satellite transmitter 170, a satellite 175, and a satellite receiver 180 provide a fast, one-way link for transferring data from satellite gateway 160 to

5 hybrid terminal 110. Each of SLIP provider 130, application server 140, and hybrid gateway 150 are connected to the Internet 128. As is well-known in the art, the Internet 128 is a "network of networks" and can be visually depicted only in general terms, as seen in Fig. 1.

10 Each of hybrid terminal 110, SLIP provider 130, application server 140, hybrid gateway 150 and satellite gateway 160 includes a processor (not shown) that executes instructions stored in a memory (not shown). Other parts of the invention also include processors that are not discussed herein, such as I/O processors,
15 etc. Preferably, hybrid terminal 110, hybrid gateway 150, and satellite gateway 160 are implemented as personal computers including an 80386/80486 based personal computer operating at least 33 MHz, but these elements can be implemented using any data processing system capable of performing the functions described
20 herein. In the described embodiment, SLIP provider 130 is a conventional SLIP provider and application server 140 is any application server that can connect to the Internet via TCP/IP.

25 As shown in Fig. 1, hybrid terminal 110 preferably includes application software 112, driver software 114, a serial port 122 for connecting hybrid terminal 110 to modem 190, and satellite interface hardware 120 for connecting hybrid terminal 110 to satellite receiver 180.

Fig. 2 shows a relationship between software in application 112, software in driver 114, serial port 122, and satellite interface 120. Application software 112 includes TCP/IP software, such as SuperTCP, manufactured by Frontier, Inc., Chameleon,

LAW OFFICES

INEGAN, HENDERSON
ARABOW, GARRETT
& DUNNER

1300 I STREET, N.W.
WASHINGTON, DC 20005
1-202-408-4000

5 manufactured by Netmanager, and IRNSS, manufactured by Spry, Inc.
The described embodiment preferably operates with the SuperTCP
TCP/IP package and, thus, uses a standard interface 212 between
the TCP/IP software 210 and driver 114. Examples of standard
10 interfaces 212 between TCP/IP software 210 and driver 114 include
the Crynson-Clark Packet Driver Specification and the
3Com/Microsoft Network Driver Interface Specification (NDIS).
Other embodiments use other standard or non-standard interfaces
between TCP/IP software 210 and driver 114.

15 As shown in Fig. 2, application software 112 also includes
well-known Internet utilities, such as FTP, and well-known user
interfaces, such as Mosaic and Gopher (shown). Application
software 112 can also include other utilities, e.g., News and
Archie (not shown).

20 The following paragraphs describe how a request from hybrid
terminal 110 is carried through the Internet 128 to application
server 140 and how a response of application server 140 is carried
back to the user at hybrid terminal 110 via the satellite link.
The operation of each subsystem will be described below in detail
in separate sections.

25 In the present invention, hybrid terminal 110 is given two IP
addresses. One IP packet address corresponds to SLIP provider 130
and is assigned by a SLIP service provider. The other IP address
corresponds to satellite interface 120 and is assigned by a hybrid
service provider. IP addresses are assigned by the SLIP and
satellite network managers and loaded into hybrid terminal 110 as
part of an installation configuration of the hybrid terminal's

5 hardware and software. These two IP addresses correspond to
completely different physical networks. SLIP provider 130 does
not "know" anything about the satellite IP address or even whether
the user is using the satellite service. If a host somewhere in
the Internet is trying to deliver a packet to the satellite IP
10 address by using the Internet routing scheme of routers, gateways,
and ARPs (Address Resolution protocol), the only way that the
packet can reach the satellite IP interface is to traverse the
satellite by being routed through satellite gateway 160.

The following example assumes that a user at hybrid terminal
15 110 desires to send a request to a remote machine, such as
application server 140, that is running FTP (File Transfer
protocol) server software. The FTP software running on
application server 140 receives file transfer requests and
responds to them in an appropriate fashion.

20 Fig. 1 shows the contents of a source field (SA) and of a
destination field (DA) of packets sent between the elements of
Fig. 1. A request for a file and a response of a file sent from
application server 140 to hybrid terminal 110 takes the following
path.

- 25 1) Within hybrid terminal 110, FTP client software 230
generates a request and passes it to TCP/IP software
210. TCP/IP software 210 places the request in a TCP
packet (see Fig. 11). Next, the TCP packet is placed in
an IP packet, having a format shown in Fig. 3. TCP/IP
software 210 places the IP packet in an Ethernet packet,
as shown in Fig. 4, and passes the Ethernet packet to

5 driver 114. This packet has a source IP address
corresponding to satellite interface 120 and a
destination IP address of application server 140.

- 10 2) In driver 114, the Ethernet header and checksum are
stripped off the packet and the IP packet is
encapsulated, or "tunneled," inside of another IP packet
and sent over serial port 122 to SLIP provider 130.

15 Fig. 5 shows a format of a tunnelled packet. Fig. 7
shows an example of a tunnelled packet. The
encapsulation adds a new IP header 530 in front of the
original packet 540 with a source address corresponding
to SLIP provider 130 and a destination address
corresponding to hybrid gateway 150.

- 20 3) SLIP provider 130 receives the IP packet, analyzes the
tunneling header and, thinking it is destined for hybrid
gateway 150, uses standard Internet routing to send the
packet to hybrid gateway 150.

- 25 4) When hybrid gateway 150 receives the packet, it strips
off the tunneling header, revealing the true header with
application server 140 as the destination. The packet
is then sent back out into the Internet 128.

- 5) Internet routing takes the packet to application server
140, which replies with the requested file and addresses.

5 the reply to the request's source IP address, i.e., the
IP address of the hybrid terminal's satellite interface
120.

10 6) In order to find the hybrid terminal's satellite
interface 120, the Internet routing protocol will send
the packet to the subnet containing a router/gateway
connected to hybrid gateway 150. When a router on the
same physical network as satellite gateway 160 and
15 hybrid gateway 150 sends out an ARP for the IP address
of satellite interface 120 (to find a physical address
of satellite interface 120), hybrid gateway 150 responds
and says "send it to me." Thus, application server 140
and the rest of the Internet 128 thinks that packets
sent to hybrid gateway 150 will reach the hybrid
20 terminal's satellite interface.

25 7) Once hybrid gateway 150 receives a reply packet from
application server 140, it sends it to satellite gateway
160. In the described embodiment, hybrid gateway 150
encapsulates the packet in a special packet format that
is used over the satellite link and uses the satellite
interface IP address to uniquely identify the satellite
packet's destination. Then hybrid gateway 150 sends the
packet over the Ethernet to satellite gateway 160.

5 8) Satellite gateway 160 broadcasts over the satellite link any packets it receives from hybrid gateway 150.

10 9) Driver 114 in hybrid terminal 110 that services satellite interface 120 scans all packets broadcast over satellite transmitter 170 looking for its satellite interface IP address in the header. Once it identifies one, it captures it, strips off the satellite header revealing the reply IP packet, and sends it to driver 114.

15 Thus, IP packets sent into Internet 128 are carried by the SLIP connection, while IP packets from the Internet 128 are carried by the satellite link. The following paragraphs describe the operation of each subsystem in more detail.

20 1. The Hybrid Terminal

25 Hybrid terminal 110 is the terminal with which the user interacts. Thus, hybrid terminal 110 includes a user interface device (not shown) such as a mouse, keyboard, etc. As shown in Fig. 1, hybrid terminal 110 includes one or more application programs 112 (including TCP/IP software 210), and driver software 114, which communicates with SLIP provider 130 through a serial port 122 and modem 190, using a driver portion 118, and which communicates with satellite receiver 180 through a satellite interface 120, using a driver portion 116.

 To TCP/IP software 210, driver 114 appears to be an Ethernet card, although driver 114 is actually connected to satellite receiver 180 (via satellite interface 120) and to SLIP provider

5 130 (via serial line 122 and modem 190). Thus, TCP/IP software
210 believes that it is communicating with a single physical
network, when it is, in reality, communicating with two physical
networks (the SLIP dialup network and a satellite network).
Ethernet is a packet switching protocol standardized by Xerox
10 Corporation, Intel Corporation, and Digital Equipment Corporation,
which is described in "The Ethernet: A Local Area Network Data
Link Layer and Physical Layer Specification," September 1980,
which is available from any of these three companies, and which is
incorporated by reference.

15 Fig. 6 is a diagram of steps performed by driver 114 of
hybrid terminal 110 of Fig. 1. As shown in Fig. 6, driver 114
receives packets of data from TCP/IP software 210 and passes them
to SLIP provider 130 via serial port 122 and modem 190. A packet
sent by application server 140 is received through satellite
20 receiver 180, passed through the satellite interface 120, passed
to the satellite driver 220, and passed to driver 114, which
passes the received packet to TCP/IP software 210.

The following paragraphs discuss two basic functions
performed by driver 114 (tunnelling and ARP handling) and discuss
25 various implementation details for the described embodiment.

A. "Tunnelling"

As discussed above, hybrid terminal 110 has two IP addresses
associated with it: one for SLIP provider 130 and one for the
satellite interface. Packets containing requests are sent from
hybrid terminal 110 to application server 140 via the Internet
128, while packets containing a reply are sent back via the

5 satellite link. Tunnelling is the method by which application server 140 is "fooled" into sending a reply to a different IP address (satellite interface 120) than that of the sender's (serial port 122).

10 A packet received by driver 114 from the TCP/IP software 219 has a source address of satellite gateway 160 and a destination address of application server 140. As shown in step 610 of Fig. 6, driver 114 removes the Ethernet header and checksum and encapsulates the IP header into an IP tunnelling header having a source address of SLIP provider 130 and a destination address of hybrid gateway 150 (see Fig. 7). As described above, at hybrid gateway 150, the tunnelling header is removed and the packet is sent back into the Internet 128 to be sent to application server 140.

20 When forming a tunnelling header, driver 114 copies all the values from the old header into the new one with the following exceptions. The source and destination addresses of the tunnelling header change, as described above. In addition, a total packet length field 510 is changed to contain the contents of length field 310 plus the length of the tunnelling header. Lastly, the driver 114 recalculates checksum 520 of the tunnelling header because some of the fields have changed.

25 B. ARP Handling

ARP (Address Resolution Protocol) is used by TCP/IP to dynamically bind a physical address, such as an Ethernet address, to an IP address. When TCP/IP finds an IP address for which it does not know a physical address, TCP/IP broadcasts an ARP packet.

5 to all nodes, expecting a response that tells TCP/IP what physical address corresponds to the IP address.

During initialization, driver 114 declares to TCP/IP software 210 that driver 114 is an Ethernet card to ensure that the packets that TCP/IP package sends are Ethernet packets and that the TCP/IP package will be prepared to receive packets at a high-rate of speed. As shown in step 620 of Fig. 6, when driver 114 detects that TCP/IP has sent an ARP packet, driver 114 creates a physical address and sends a reply packet to TCP/IP software 210. The contents of the physical address are irrelevant, because driver 114 strips off the Ethernet header on packets from TCP/IP before the packets are sent to SLIP provider 130.

C. Other Functions

As shown in step 630 of Fig. 6, packets received by driver 114 from satellite receiver 180 (via satellite driver 114) are merely passed to TCP/IP software 210. The following paragraphs discuss implementation details for the described embodiment.

In a preferred embodiment, TCP/IP software 210 (e.g., Frontier's SuperTCP) sends an ACK (acknowledge) for every packet it receives, even though this action is not required by the TCP/IP protocol. In this situation, many packets compete for the slow link to SLIP provider 130. In TCP/IP, the ACK scheme is cumulative. This means that when a transmitter receives an ACK stating that the receiver has received a packet with sequence number N, then the receiver has received all packets with sequence numbers up to N as well, and there is no reason why every packet needs to be ACK'ed.

5 Fig. 8 is a flowchart of steps performed in a preferred
embodiment by driver 114 of hybrid terminal 110. Fig. 11 is a
diagram showing a TCP packet format. Fig. 11 includes a sequence
number field 1102, an acknowledgment (ACK) number field 1104, and
a checksum field 1106. In step 810 of Fig. 8, driver 114 receives
10 an ACK packet with sequence number N from TCP/IP software 210.
The packet is queued along with other packets waiting to be sent
to SLIP provider 130. In step 820 driver 114 checks to determine
whether there is a "run" of sequential packets waiting to be sent.
If so, in step 830, driver 114 deletes ACK packets for the same
15 TCP connection that have sequence numbers in the run from the
queue and sends an ACK only for the highest sequence number in the
run. This action alleviates the bottleneck caused by the
relatively slow modem speeds.

20 Serial port 122 provides a physical connection to modem 190
and, through it, to the terrestrial network via a SLIP protocol as
described below in connection with SLIP provider 130. Serial data
is sent and received through an RS-232 port connector by a UART
(Universal Asynchronous Receiver Transmitter), such as a U8250,
which has a one byte buffer and is manufactured by National
25 Semiconductor, or a U16550, which has a 16 byte buffer and is also
manufactured by National Semiconductor.

The invention preferably operates under the DOS operating
system and Windows, but also can operate under other operating
systems.

Satellite driver software 220 receives packets from satellite
180, and passes them to driver 114 using a DOS call. Thus, the

5 two physical links are combined within driver 114 and the
existence of two physical links is transparent to TCP/IP software
210. Satellite driver 220 scans all packets transmitted over the
satellite channel for a packet with a header corresponding to the
IP address of the satellite interface 122, performs some error
10 detection and correction on the packet, buffers the received
packet, and passes the packet to driver 114 using a DOS call,
e.g., IOCTL-output-cmd(). Driver 114 copies data from satellite
driver 220 as quickly as possible and passes it to TCP/IP software
210.

15 As discussed above, TCP/IP software 210 is fooled into
thinking that it is connected to an Ethernet network that can send
and receive at 10 Mbps. This concept is helpful on the receive
side because data from the satellite is being received at a high
rate. On the transmit side, however, modem 190 is not capable of
20 sending at such a high rate. In addition, TCP/IP software 210
sends Ethernet packets to driver 114, i.e., an IP packet is
encapsulated into an Ethernet packet. Because SLIP provider 130
expects IP packets, driver 114 must strip the Ethernet header
before the packet is sent to SLIP provider 130.

25 As described above in connection with Fig. 8, Driver 114 also
includes a transmit and receive queue. As data is received from
TCP/IP software 210 and received from the satellite driver 220, it
is buffered within the queue. When the queue is full, e.g., when
TCP/IP is sending packets faster than modem 190 can send them,
driver 114 drops the packets and returns an error so that TCP/IP
software 210 will decrease its rate of transmission.

5 In a first preferred embodiment, a SLIP connection is initiated with an automatic logon procedure. In another preferred embodiment, driver 114 executes instructions to allow a user to perform a SLIP logon manually.

10 Because TCP/IP software 210 preferably is configured to talk to Ethernet and it is desirable to receive the largest packet size possible, driver 114 configures TCP/IP so that the MTU (Maximum Transmission Unit) of the network is as large as possible, e.g., 1500 bytes. Some SLIP providers 130 have a smaller MTU, e.g., 512 bytes. To handle the disparity in size, driver 114 segments large
15 packets received from TCP/IP software 210 into segments the size of the SLIP MTU. Once a packet is segmented, it is reassembled in hybrid gateway 150. Only the tunnelling header is copied as the header of the segments.

20 2. The SLIP Provider

SLIP provider 130 performs the function of connecting hybrid terminal 110 to the Internet 128. As described above, other protocols, such as PPP, could also be used to perform the connecting function. SLIP server 130 receives SLIP encoded IP packets from modem 190, uncodes them, and forwards them to hybrid
25 gateway 150 via the Internet 128.

In its most basic form, SLIP provider 130 delimits IP packets by inserting a control character hex 0xC0 between them. To insure that a data byte is not mistaken for the control character, all outgoing data is scanned for instances of the control character, which is replaced by a two character string. The SLIP protocol is described in detail in J. Romkey, "A Nonstandard for Transmission

5 of IP Datagrams over Serial Lines: SLIP," RFC 1055, June 1988, pp. 1-6, which is incorporated by reference.

3. The Application Server

Application server 140 is a computer system running any combination of known application programs available on the Internet using the TCP/IP protocol suite. For example, application server 140 may be transferring files to requesting users via FTP. Although hybrid terminal 110 actually has two IP addresses (a serial port address and an address for the satellite interface), the software executing on application server 140 thinks that it is receiving requests over the satellite network and sending responses over the satellite network. Hybrid terminal 110 is completely transparent to application server 140.

4. The Hybrid Gateway

Although only one hybrid terminal 110 is shown in Fig. 1, the invention can include a plurality of hybrid terminals 110. Preferably, all packets sent from all hybrid terminals 110 pass through hybrid gateway 150 to get untunnelled. Thus, hybrid gateway 150 is a potential system bottleneck. Because of this potential bottleneck, the functions of hybrid gateway 150 are as simple as possible and are performed as quickly as possible. Hybrid gateway 150 also has good Internet connectivity to minimize the accumulated delay caused by packets waiting to be processed by hybrid gateway 150.

A. Untunnelling

Fig. 9 is a diagram of steps performed by hybrid gateway 150 of Fig. 1. In step 910, hybrid gateway 150 receives a tunnelled

5 packet having a format shown in Fig. 5. Hybrid gateway 150
"untunnels" the packet by stripping off the tunnelling header and
passes the packet back to the Internet.

As described above, packets are sometimes broken into
segments when they are sent in order to accommodate a small MTU of
10 SLIP provider 130. Packets may also be segmented as they pass
through other elements of the net having small MTUs. For
fragmented packets, only the tunnelled header is copied into the
header of each segment. Hybrid gateway 150 stores fragmented
packets in a memory (not shown) and reassembles them in order
15 before untunnelling the original packet and passing it to the
Internet. Preferably, a "time to live" value is assigned to each
packet when it is sent by driver 114 and if all segments do not
arrive before a time to live timer expires, the packet is
discarded.

20 B. ARP Responding

Preferably, satellite gateway 160 is on a same physical
network as hybrid gateway 150. As shown in step 920 of Fig. 2,
when a router on the same physical network as satellite gateway
160 and hybrid gateway 150 sends out an ARP for the IP address of
25 satellite gateway 160 (to find a physical address of satellite
gateway 160), hybrid gateway 150 responds and says "send it to
me." Hybrid gateway 150 needs to intercept packets intended for
satellite gateway 160 because it needs to encapsulate packets for
satellite gateway 160 as follows.

LAW OFFICES

INEGAN, HENDERSON
ARABOW, GARRETT
& DUNNER

1300 I STREET, N. W.
WASHINGTON, DC 20005
1-202-408-4000

C. Satellite Packetizing

The following paragraphs describe how packets travel from application server 140 through hybrid gateway 150 and to satellite gateway 160. The following explanation is given by way of example and is not intended to limit the scope of the present invention.

10

As shown in step 930 of Fig. 9, hybrid gateway 150 encapsulates replies from application server 140 into a satellite packet format. Fig. 10 is a diagram showing a format of a satellite packet sent to satellite gateway 160 of Fig. 1. A satellite packet includes the data 1010 of an original IP packet and two headers added by hybrid gateway 150.

15

Satellite gateway 160 expects IP packets to be encapsulated first in a special satellite packet and then within an LLC-1 IEEE 802.2 link level control, type 1 packet. Satellite header 1020 identifies the downlink and contains a sequence number and the packet length. An LLC-1 header 1030 preferably is used to send the packet to satellite gateway 160, in an Ethernet LAN. Hybrid gateway 150 prepares packets for satellite gateway 160 by appending headers 1020 and 1030 to the front of an IP packet 1010.

20

The receiver in hybrid terminal 110 does not receive the LLC-1 header 1030. Hybrid terminal 110 identifies packets intended for it by checking a least significant byte in the satellite IP address. Thus, a six byte satellite destination address is determined by reversing an order of bytes of the satellite IP address for hybrid terminal 110 and then padding the rest of the address with zeroes.

25

5 5. The Satellite Gateway

Satellite gateway 160 can include any combination of hardware and software that connects satellite transmitter 170 to hybrid gateway 150. Satellite transmitter 170 and satellite receiver 180 can be any combination of hardware and software that allows data to be transmitted by satellite transmitter 170 and received by satellite receiver 180, and to be input to hybrid terminal 110. For example, satellite gateway 160 preferably is a personal computer with a high-speed Ethernet connection to hybrid terminal 110. When satellite gateway 160 receives a packet from hybrid gateway 150, it sends it over the satellite link.

Satellite communication may be effected by, for example, the Personal Earth station manufactured by Hughes Network Systems Corp. In a preferred embodiment, a one-way version of the Personal Earth Station is used. Another embodiment uses a satellite communication system manufactured by Comstream. Yet another embodiment uses a system that allows hybrid terminal 110 to be connected directly to satellite receiver 180 via Hughes Network Systems' DirectPC product. The DirectPC satellite interface card is described in "DirectPC, Phase A Data Sheet," dated June 7, 1993, which is attached as Appendix A, which is incorporated in and constitutes a part of this specification, and which is incorporated by reference.

At the downlink, satellite receiver 180 includes a 0.6 meter receive-only antenna receiving HDLC encapsulated LAN packets. Satellite interface 120 includes rate 2/3 Viterbi/Reed-Solomon concatenated forward error correction.

5 Although only one hybrid terminal 110 and one application
server 140 are shown in Fig. 1, the invention can include a
plurality of hybrid terminals 110 and/or a plurality of
application servers 140. Preferably, all packets sent from all
application servers 140 to a hybrid interface 110 pass through
10 satellite gateway 160. Thus, satellite gateway 160 is a potential
system bottleneck. Because of this potential bottleneck, the
functions of satellite gateway 160 are as simple as possible and
are performed as quickly as possible.

c. Protocol Spoofing

15 TCP/IP protocol specifies that only a predetermined number of
packets can be outstanding during transmission, i.e., that only a
limited number of packets can be sent before an ACK
(acknowledgment) is received. The high bandwidth and long delays
incurred in sending packets to an orbiting satellite and back
20 means that at any given time, a lot of packets are "in the pipe"
between transmitter and receiver.

 When using conventional TCP/IP protocol, application server
140 sends a predetermined number of packets in accordance with a
predetermined window size, and then waits to receive ACKs over the
modem link before sending additional packets. The purpose of
25 windowing is to limit a number of packets that must be re-sent if
no ACK is received and to provide flow control, e.g., to prevent
sending packets faster than they can be received. The packets
that have not been ACK'ed are stored in a memory so that they can
be re-sent if no ACK is received.

5 In a preferred embodiment of the present invention, hybrid gateway 150 "spoofs" application server 140 to improve the throughput over the satellite link. Specifically, hybrid gateway 150 sends an ACK to application server 140, even though a corresponding packet may not have been received by hybrid terminal 110 via the satellite at the time.

10 Fig. 12 is a ladder diagram showing packets sent from application server 140 to hybrid gateway 150 and from hybrid gateway to hybrid terminal 110 through the satellite link. Fig. 12 is not drawn to scale. In Fig. 12, application server 140 sends a message #1 to hybrid gateway 150. The propagation time for this transmission is relatively short. Hybrid gateway 150 immediately creates an ACK packet and sends it to application server 140. Hybrid gateway also sends packet #1 to hybrid terminal 110 through the satellite link. This transmission has a long propagation delay. When hybrid terminal 110 receives the packet, it sends an ACK #1 back to hybrid gateway 150 (e.g., using the tunnelling mechanism described above). In a system that does not use tunnelling, hybrid gateway 150 needs to intercept the ACK packets from hybrid terminal 110.

25 Figs. 13(a) through 13(e) are flowcharts of steps performed by hybrid gateway 150 of Fig. 1 during protocol spoofing. In step 1302 of Fig. 13(a), hybrid gateway 150 receives a packet from application server 140 indicating that a new connection is being formed between application server 140 and hybrid terminal 110. In step 1304, hybrid gateway 150 sets up a queue or similar data

5 structure in memory to save un-ACK'ed packets for the new connection. Fig. 13(b) show corresponding steps performed by hybrid gateway 150 when the connection is closed. Hybrid gateway 150 receives a packet indicating the closure in step 1306 and deletes the queue and saved values for the connection in step 1308.

10 In step 1310 of Fig. 13(c), hybrid gateway 150 fails to receive an ACK for a packet number X from hybrid terminal 110 before an end of a predetermined timeout period. Hybrid gateway 150 maintains a timer for each un-ACK'ed packet. At the end of the predetermined period, hybrid gateway 150 retransmits a packet
15 corresponding to the expired timer. In step 1312, hybrid gateway 150 re-sends packet number X, which it previously saved in the memory queue for this connection (see Fig. 13(d) below).

In step 1314 of Fig. 13(d), hybrid gateway 150 receives a packet from application server 140. In step 1316, hybrid gateway
20 150 sends the received packet to satellite gateway 160, where it is transmitted over the satellite link, and saves the packet in case it needs to be retransmitted (see Fig. 13(c)). Hybrid gateway 150 then creates an ACK packet to send to application server 140 in step 1318. The created ACK packet incorporates a format
25 shown in Fig. 11. Hybrid gateway 150 creates an ACK number for field 1104. The ACK number is determined as follows:

Hybrid gateway 150 saves the following information for each connection:

1) Send sequence number - a highest in-sequence sequence number of packets sent by application server 140 over the connection.

LAW OFFICES

NECAN, HENDERSON
FARABOW, GARRETT
& DUNNER

1300 I STREET, N. W.
WASHINGTON, DC 20005
1-202-408-4000

5 2) ACK sequence number - the ACK sequence number from the most recent packet sent by hybrid terminal 110 over this connection.

 3) ACK window size - the window size from the most recent packet from hybrid terminal 110 over this connection.

10 4) ACK number - the ACK sequence number that is relayed to application server 140. The ACK number is set to:

 minimum(send sequence number, ACK sequence number +
 spoofed window size - ACK window size).

15 5) spoofed window size - predetermined maximum number window size to be allowed on this connection.

 When hybrid gateway 150 inserts the ACK number in the packet, it also calculates the packet's checksum 1106.

20 In step 1320 of Fig. 13(e), hybrid gateway 150 receives an ACK packet over the modem link from hybrid terminal 110. In step 1322, hybrid gateway 150 removes from the queue the packet for which the ACK was received. Because an ACK was received, the packet does not need to be re-sent. In the TCP/IP protocol, a
25 packet containing an ACK may or may not contain data. Hybrid gateway 150 edits the received packet to replace the packet's ACK number 1104 with a "spoofed" ACK number in step 1326. The spoofed ACK number is determined in the same way as the ACK number in step 1318 of Fig. 13(d). When hybrid gateway 150 substitutes the spoofed ACK number 1104 in the packet, it also recalculates the packet's checksum 1106 in step 1326.

5 In step 1328, hybrid gateway 150 forwards the received ACK packet to application server 140. Application server 140 may simply disregard the packet if it contains an ACK and no data. In another embodiment, hybrid gateway 150 simply discards a packet received from hybrid terminal 110 that contains an ACK, but no data.

10 If the connection goes down, either explicitly or after a predetermined period of time, hybrid gateway 150 deletes the saved packets for the connection.

d. Summary

15 In summary, the present invention allows a personal computer to send messages into the Internet using a conventional dial-up link and to download data from the Internet using a high-speed one-way satellite link. In a preferred embodiment, the invention uses a conventional SLIP provider to connect to the Internet and uses a commercial software TCP/IP package that has a standard driver interface. A spoofing protocol compensates for the long propagation delays inherent to satellite communication.

20 Other embodiments will be apparent to those skilled in the art from consideration of the specification and practice of the invention disclosed herein. It is intended that the specification and examples be considered as exemplary only, with a true scope of the invention being indicated by the following claims.